STRONG VALID INEQUALITY CONSTRAINTS FOR ARCHITECTURAL LAYOUT DESIGN OPTIMIZATION

KEATRUANGKAMALA K.
Department of Computer-Aided Architectural Design,
Faculty of Architecture, Rangsit University, Thailand
kamolkeat@hotmail.com

AND

NILKAEW P.
Faculty of Architecture,
Chiang Mai University, Thailand
tuan@mail.cmu.ac.th

Keyword: Optimization, Mixed integer programming, Strong valid inequality, Architectural layout design.

Abstract. In the past decades, many attempts have been made to solve the challenging architectural layout design problem such as non-linear programming and evolutionary algorithm (Michalek and Papalambros, 2002). The Mixed Integer Programming (MIP) (Kamol and Krung, 2005) was recently developed to find the global optimal solution. However, the problem can be shown to belong to the class of NP-hard problem (Michalek and Papalambros, 2002). Hence, only the small instances of the problem can be solved in a reasonable time. In order to deal with large problem sizes, this paper utilizes the strong valid inequalities (George and Laurence). It cut off the infeasible points in the integral search space by formulated the disconnected constraints involved with line configurations of three rooms. It is shown to significantly increase the computational speed to more than thirty percents. This exhibits the practical use of the MIP formulation to solve the medium size architectural layout design problems.

1. Introductions

Architectural layout design is particularly interesting because in addition to common engineering objectives such as performance, architectural design is especially concerned with the quantifiable aspects of architectural floorplan and usability qualities of a layout which are generally more difficult to describe formally. By which, architectural layout be distinguished with many researches on spatial configuration includes component packing, route path planning and VLSI design (Rabbat, G., 1988). Nevertheless, many researches attempts to automate the process of architectural layout design started over two decades ago. Researchers have used several representations and solution techniques to describe and solve this problem. But architectural layout design is not easily dealt with, for example, the medium-sized problem presents an unattainable solution time.

In this research, the automated layout design tool presented here is contrasted with other research attempts to automate layout design. The distinct MIP formulation (Kamol and Krung, 2005) proposes the novel approach to guarantee the optimal design based on objective and subjective preferences. Furthermore, the methodologies to speed the solution time are suggested. Additional mathematical formulation called Strong Valid Inequality has been advised to the research. Previously, the research in Optimizing Architectural Layout Design Via Mixed Integer Programming (Kamol and Krung, 2005) published in CAADFutures2005 formulating the optimal architectural layout design problem as the multiobjective mixed integer programming model solved by the MIP solver. However, the instances of the optimal architectural layout design problem are unattainable for the medium to large problem sizes. To reduce the computational time, the strong valid inequality constraints have been adopted and added which reduced the feasible region based on the two decision binary variables p, q from the previous research. The improvement model used eight disconnected room connectivity reducing the feasible region of the LP relaxation. The experiment showed to have a significant reduction of the original solution time.

2. Methodologies

2.1 STRONG VALID INEQUALITY

To reduce the solution of the MIP using the mathematical model, we incorporate the strong valid inequality which is based on the tighter feasible region. By which, the search space has been cut off the non-integral feasible region but it still maintains all feasible integral points. We add more inequality constraints to an original formulation called the strong valid inequality constraints. The notions of the strong valid inequality can be defined in the mathematical term as follows.

Given the IP (Integer programming problem) as

$$\text{(IP)} \qquad \max\{\, c^{T}x : x \in X \,\} \qquad (1)$$

$$X = \{\, x : Ax \leq b \,,\, x \in Z^{+}_{n} \,\} \qquad (2)$$

Note that conv(X) = $\{\, x : Ax < \bar{b},\, x \in Z^{+}_{n} \,\}$ for some matrix A and some vector b (i.e., conv(X) is a polyhedron). The LP relaxation of the IP is

$$\text{(LP)} \qquad \max\{\, c^{T}x : Ax \leq \bar{b} \,,\, x > 0 \,\} \qquad (3)$$

The inequality $\pi^{T}x < \pi_0$ or (π, π_0) is called a valid inequality for X if $\pi^{T}x \leq \pi_0$ for all $x \in X$.

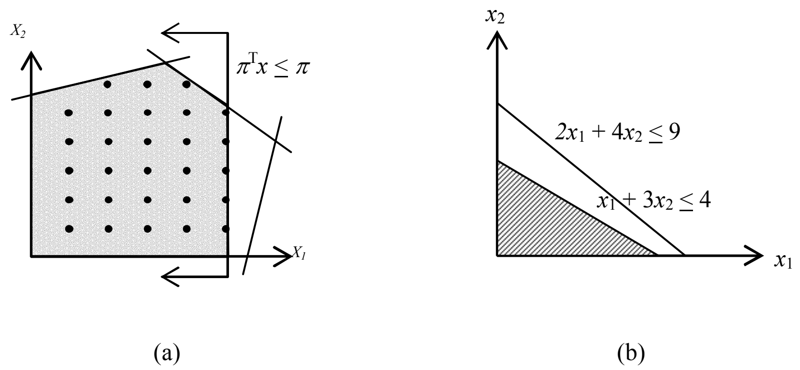


Figure 1. (a) the conceptual strong valid inequality cut $\pi^{T}x \leq \pi$ is used to reduce the feasible region and (b) example, the strong valid inequality constraint $x_1+3x_2 \leq 4$ dominates $2x_1+4x_2 \leq 9$

2.2 CONNECTED CONNECTION

Figure 1 illustrates the half space which is defined as the strong valid inequality. The new feasible region of the LP relaxation is strictly smaller than the LP relaxation of the original one. The smaller convex set was generated by adding eight directional connectivities. Three ordering subscripted set of room connectivity *i*, *j* and *k* of binary variables *p*, *q* are adopted. The strong valid inequalities uses the disconnected constraints of the room *i* and *k* to construct the negated constraints for each consecutive direction (above, left, right and below) of room *i* and *j* (see the figure 1b). The following inequality constraints present the unconnected direction of room *i* and *k* following each connection of room *i* and *j*.

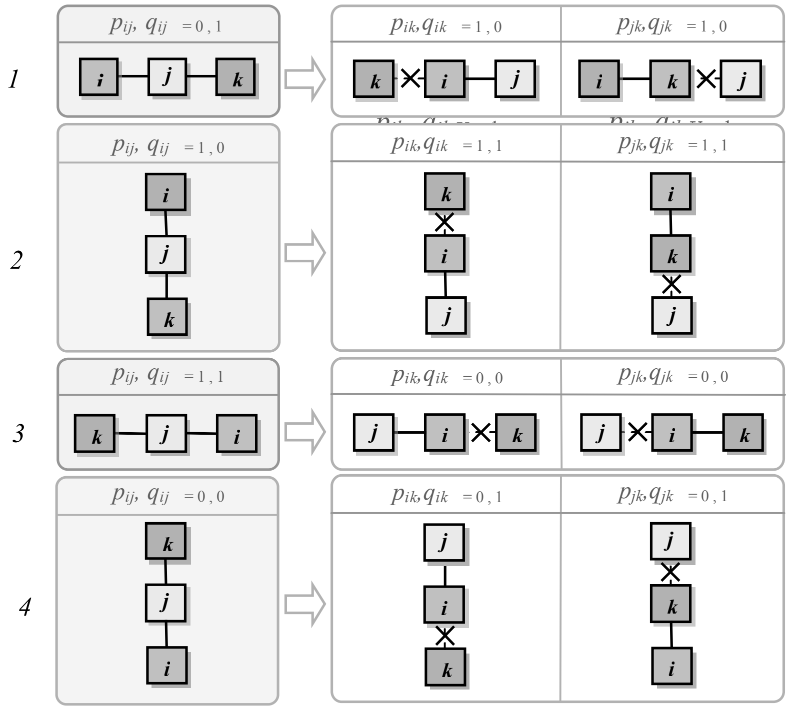


Figure 2. Show the concept of disconnected constraints, in case 1: If the connection of p_{ij},q_{ij} is 0,0 then the connection of $p_{ik}q_{ik}$ and $p_{jk}q_{jk}$ can not be 1,0. Similarly, case 2,3 and 4 shows the disconnected configuration on the right side.

3. Experiments

3.1 DISCONNECTED CONSTRAINTS

The improvement model proposed the mathematical process to solve the architectural layout design. By which, we achieved the solution time reduction using the disconnected constraint. The three consecutive room connectivity for each direction has been converted to the optimization formulae that fit to the initial MIP model. The following strong valid inequality constraints present the unconnected direction of room *i* and *k* following each connection of room *i* and *j* (The disconnected constraint has been shown as below).

$1 - (1-p_{ik}) + (0-q_{ik})$	$<=$ Width*$(p_{ij} + q_{ij})$	*i* to the left of *j*,	$p_{ij}=0$, $q_{ij}=0$
$1 - (1-p_{jk}) + (0-q_{jk})$	$<=$ Width*$(p_{ij} + q_{ij})$	*i* to the left of *j*,	$p_{ij}=0$, $q_{ij}=0$
$1 - (1-p_{ik}) - (1-q_{ik})$	$<=$ Width*$(1 + p_{ij} - q_{ij})$	*i* above *j*,	$p_{ij}=0$, $q_{ij}=1$
$1 - (1-p_{jk}) - (1-q_{jk})$	$<=$ Width*$(1 + p_{ij} - q_{ij})$	*i* above *j*,	$p_{ij}=0$, $q_{ij}=1$
$1 + (0-p_{ik}) + (0-q_{ik})$	$<=$ Width*$(1 - p_{ij} + q_{ij})$	*i* to the right of *j*,	$p_{ij}=1$, $q_{ij}=0$
$1 + (0-p_{jk}) + (0-q_{jk})$	$<=$ Width*$(1 - p_{ij} + q_{ij})$	*i* to the right of j,	$p_{ij}=1$, $q_{ij}=0$

$1 + (0-p_{ik}) - (1-q_{ik})$	$<= \text{Width}*(2 - p_{ij} - q_{ij})$	i below j,	$p_{ij}=1$, $q_{ij}=1$
$1 + (0-p_{jk}) - (1-q_{jk})$	$<= \text{Width}*(2 - p_{ij} - q_{ij})$	i below j,	$p_{ij}=1$, $q_{ij}=1$

where: *p, q* are the decision binary variables from the previous research.
Width is the width of the layout boundary.

3.2 COMPARISONS

The experiment used the non-commercial linear programming and MIP solver, GLPK (GNU Linear Programming Kit) that was developed by Moscow Aviation Institute (Russia). This research improved the problem formulae using the disconnected constraints. The results of the problem present the global optimal solution. (see the following table).

Numbers	Avg conventional MIP			Avg strong valid inequalities MIP		
of room	Time	Loops	Memory used	Time	Loops	Memory used
4	1.1	2619	0.4M	0.7	1644	0.4M
5	62.7	46585	1.1M	14.7	22967	0.7M
6	3854	2223416	6.8M	397.8	376852	2.7M

Note: all experiments were tested on Pentium 1 GHz and 256 MB of RAM.

The mathematical reduction uses of the strong valid inequalities reduce the feasible integer region by enforcing the rearrangement of the problem formulation time reduction (see the figure 3) of the disconnected constraints. This methodology has been shown to have a significant gain of more than thirty percents computational time.

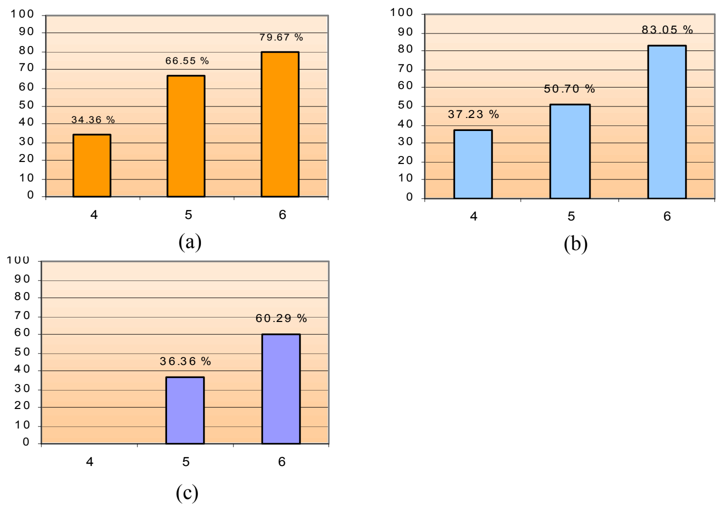


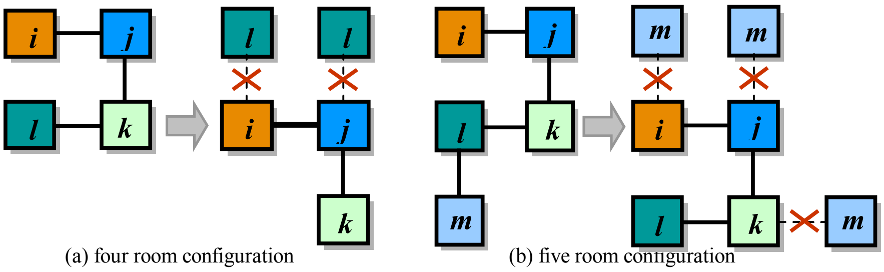
Figure 3. (a) percentage of timing comparison between avg. conventional MIP and strong valid inequalities MIP, (b) percentage of iterative comparison and (c) percentage of memory used comparison.

4. Conclusions

This research shows the practical formulation of MIP to solve the architectural layout design based on the strong valid inequalities. The instance of the problem presents the significant time reduction more than thirty percents achieved.

FUTURE WORKS

To tackle the medium-sized of room, we propose the extending of the disconnected constraint from three room configurations. Moreover, revising the branch and bound algorithm and developing of the computer software which embedded the possible solution of room configurations will be speed up the computational performance.



(a) four room configuration (b) five room configuration

Figure 4. based on the disconnected constraint, four (a) and five (b) room configurations has been applied to strong valid inequality.

Acknowledgements

I am grateful for the comments I have received from my advisor at Chulalongkorn University, Prof. Krung Sinapironsaran in particular for this valuable work and suggestions as well as my advisee at Rangsit University, Mr. Piyaboon Nilkeaw in particular for represention of my research works at CAADRIA 2006.

References

Keatruangkamala, K. and Sinapiromsaran, K. 2005. Optimizing Architectural Design via Mixed Integer Programming. *Proceeding in CAAD Futures 2005* 11: 175-184.

Michalek, J. and Papalambros, P.Y. 2002. Interactive layout design optimization. *Engineering Optimization* 34(5): 461-184.

Medjodoub, B., and Yannon, B. 2000. Separating Topology and Geometry in space planning. *Computer-Aided Design* 32: 39-61.

Jo, J.H., and Gero, J.S. 1998. Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering* 12(3):149-162.

Jo, J.H., and Gero, J.S. 1998. Machine learning in design using genetic engineering-based genetic algorithms. *InductrialKnowledge Management*. Springer, London.

Ignizio, J. P. and Cavalier, T. M. Linear programming. *Prentice hall international, inc*. New Jersey.

George, L. N. and LAURENCE, A. W. Integer and combinatorial optimization. *Wiley-Interscience Publication*. New York. A

Rabbat, G. 1988. VLSI and AI are getting closer. *IEEE circuits and Devices Magazine*, Vol. 1, 1988.

George, L. N. and LAURENCE, A. W. Integer and combinatorial optimization. *Wiley-Interscience Publication*. New York. A